

Reweighted LP Decoding for LDPC Codes

M. Amin Khajehnejad* Alexandros G. Dimakis† Babak Hassibi* William Bradley+

*California Institute of Technology

†University of Southern California

+Lyric Semiconductors Inc.

Abstract—We introduce a novel algorithm for decoding binary linear codes by linear programming. We build on the LP decoding algorithm of Feldman et al. and introduce a post-processing step that solves a second linear program that reweights the objective function based on the outcome of the original LP decoder output. Our analysis shows that for some LDPC ensembles we can improve the provable threshold guarantees compared to standard LP decoding. We also show significant empirical performance gains for the reweighted LP decoding algorithm with very small additional computational complexity.

I. INTRODUCTION

Linear programming (LP) decoding for binary linear codes was introduced by Feldman, Karger and Wainwright [2]. The method is based on solving a linear-programming relaxation of the integer program corresponding to the maximum likelihood (ML) decoding problem. LP decoding is connected to message-passing decoding [3], [4], and graph covers [5], [6] and has received substantial recent attention (see e.g. [6], and [7]).

As with the work described here, a related line of work has studied various improvements to either standard iterative decoding [8], [9] or to LP decoding via nonlinear extensions [10] or loop corrections [11].

The practical performance of LP decoding is roughly comparable to min-sum decoding and slightly inferior to sum-product decoding. In contrast to message-passing decoding, however, the LP decoder either concedes failure on a problem, or returns a codeword along with a guarantee that it is the ML codeword, thereby eliminating any undetected decoding errors.

The main idea of this paper is to add a second LP as a post-processing step when original LP decoding fails and outputs a fractional pseudocodeword. We use the difference between the input channel likelihood and the pseudocodeword coordinate to find a measure of disagreement or unreliability for each bit. We subsequently use this unreliability to bias the objective function and re-run the LP with the reweighted objective function. The reweighting increases the cost of changing reliable bits and decreases the cost for unreliable bits. We present an analysis that can show that the provable BSC recovery thresholds improve for certain families of LDPC codes. We stress that the actual thresholds, even for the original LP decoding algorithm remain unknown. Our analysis only establishes that the obtainable lower bounds on the

fraction of recoverable errors are improved compared to the corresponding bounds for LP decoding. It is possible, however, that this is just an artifact of the lower bound techniques and that the true threshold is identical for both algorithms. In any case, the empirical performance gains we observe in our preliminary experimental analysis seem quite substantial.

A central idea in our analysis is a notion of robustness to changes in the BSC bit-flipping probability. This concept was inspired by a similar reweighted iterative ℓ_1 minimization idea for compressive sensing [20], [19]. We note that the reweighting idea of this paper involves changing the objective function of the LP and different from the reweighted max-product algorithm [12].

II. BASIC DEFINITIONS

A vector x in \mathbb{R}^n is called k -sparse if it has exactly k nonzero entries. The support set of a sparse vector x means the index set of its nonzero entries. If x is not sparse, the k -support set of x is defined as the index set of the maximum k entries of x in magnitude. We use $\|x\|_p$ to denote the ℓ_p norm of a vector x for $p \geq 0$. In particular $\|x\|_0$ is defined to be the number of nonzero of entries in x . For a set S , cardinality of S is denoted by $|S|$ and if $S \subset \{1, 2, \dots, n\}$, then x_S is the sub-vector formed by those entries of x indexed in S . Also the complement set of S is denoted by S^c . The rate of a linear binary code \mathcal{C} is denoted by R , and the corresponding parity check matrix is $H \in \mathbb{F}^{m \times n}$, where n is the length of each codeword and $m = Rn$. The factor graph corresponding to \mathcal{C} is shown by $\mathcal{G} = (X_v, X_c, \mathcal{E})$, where X_v and X_c are the sets of variable nodes and check nodes respectively and \mathcal{E} is the set of edges. If the graph is regular on either side, d_v and d_c denote the degree of variable and check nodes respectively. The girth of a graph \mathcal{G} is defined to be the size of the smallest cycle in \mathcal{G} .

III. BACKGROUND

In this section, we will review the problem of linear programming decoding for linear codes. Suppose that \mathcal{C} is a memoryless channel with binary input and an output alphabet \mathcal{Y} , defined by the transition probabilities $P_{Y|X}(y|x)$. For a received symbol y , the likelihood ratio is defined as $\log(\frac{P_{Y|X}(y|x=0)}{P_{Y|X}(y|x=1)})$, where x is the transmitted symbol. Now if a codeword $x^{(c)}$ of length n from the linear code \mathcal{C} is transmitted through the channel, and an output vector $x^{(r)}$ is received, a maximum likelihood decoder can be used to estimate the transmitted codeword by finding the *most likely*

This work was supported in part by the National Science Foundation under grants CCF-0729203, CNS-0932428 and CCF-1018927, by the Office of Naval Research under the MURI grant N00014-08-1-0747, by Caltech's Lee Center for Advanced Networking and by DARPA FA8750-07-C-0231.

transmitted input codeword. Let γ_i be the likelihood ratio assigned to the i th received bit $x_i^{(r)}$, and γ be the likelihood vector $\gamma = (\gamma_1, \dots, \gamma_n)^T$. The ML decoder can be formalized as follows

$$\begin{aligned} \text{ML decoder:} \quad & \text{minimize } \gamma^T x \\ & \text{subject to } x \in \text{conv}(\mathcal{C}), \end{aligned} \quad (1)$$

Where $\text{conv}(\mathcal{C})$ is the convex hull of all the codewords of \mathcal{C} in \mathbb{R}^n . The linear program (1) solves the ML decoding problem by virtue of the fact that the objective $\gamma^T x$ is minimized by a corner point (or vertex) of $\text{conv}(\mathcal{C})$, which is a codeword. (In fact, the vertices of $\text{conv}(\mathcal{C})$ are all the codewords of \mathcal{C} .) In a linear program, the polytope over which the optimization is performed is described by linear inequalities describing the facets of the polytope. It turns out that problem (1) is NP-hard, since $\text{conv}(\mathcal{C})$ has an exponential number of facets, as the code length n grows, and cannot be described efficiently. In [1], Feldman noted that a relaxation of (1) can be done by replacing the polytope $\text{conv}(\mathcal{C})$ with a new polytope \mathcal{P} that has much fewer, often only polynomially, number of facets, contains $\text{conv}(\mathcal{C})$ and retains the codewords of \mathcal{C} as its vertices. One way to construct \mathcal{P} is the following. If the parity check matrix of \mathcal{C} is the $m \times n$ matrix H and if h_j^T is the j -th row of H , then

$$\mathcal{P} = \cap_{1 \leq j \leq m} \text{conv}(\mathcal{C}_j) \quad (2)$$

Where $\mathcal{C}_j = \{x \in \mathbb{F}^n \mid h_j^T x = 0 \bmod 2\}$. As mentioned earlier, with this construction, all codewords of \mathcal{C} are vertices of \mathcal{P} . However, \mathcal{P} has some additional vertices with fractional entries in $[0, 1]^n$. A vertex of the polytope \mathcal{P} is called a *pseudo-codeword*. Moreover, if a pseudo-codeword is integral, i.e., if it has 0 or 1 entries, then it is definitely a codeword. The LP relaxation of (1) can thus be written as:

$$\begin{aligned} \text{LP decoder:} \quad & \text{minimize } \gamma^T x \\ & \text{subject to } x \in \mathcal{P} \end{aligned} \quad (3)$$

The number of facets of \mathcal{P} is exponential in the maximum weight of a row of H . So, for LDPC codes where each row of H has a small (often constant) number of 1's, \mathcal{P} has a polynomial number of facets, and solving (3) requires only polynomial complexity.

For binary symmetric channels, (3) has another useful interpretation. In this case, rather than minimize $\gamma^T x$ it turns out that one can alternatively minimize the Hamming distance between the output of the channel $x^{(r)}$ and the individual codewords $x \in \mathcal{C}$. Using the fact that the LP relaxation with \mathcal{P} relaxes the entries of x from $x_i \in \{0, 1\}$ to $x_i \in [0, 1]$, we may replace the Hamming distance with the ℓ_1 distance $\|x - x^{(r)}\|_1$. This implies that the decoder (3) is equivalent to

$$\begin{aligned} \text{BSC-LP decoder:} \quad & \text{minimize } \|x - x^{(r)}\|_1 \\ & \text{subject to } x \in \mathcal{P} \end{aligned} \quad (4)$$

The above formulation can be interpreted as follows. For a received output binary vector $x^{(r)}$, the solution to the LP

decoder is basically the closest (in the ℓ_1 distance sense) pseudo-codeword to $x^{(r)}$.

Linear programming decoding was first introduced by Feldman et al. [1], [2]. A later seminal result of Feldman et al. [13] proved that for random expander codes, LP decoding can correct a constant fraction of errors. A fundamental lemma in [2] and used in the results therein, is that the LP polytope \mathcal{P} is the same polytope from the view point of every codeword, and therefore for the analysis of LP decoding, it can be assumed without loss of generality that the transmitted codeword is the all zero codeword. The theoretical results of [13] were based on a dual witness argument, i.e. a feasible set of variables that set the dual of LP equal to zero. The achieved bound was, however, considerably smaller than the empirical recovery threshold of LP decoder in practice. A later probabilistic analysis of LP decoding by Daskalakis et al. [14] improved upon those bounds for random expander codes, by using a different dual witness argument, and by considering a *weak* notion of LP success rather than the *strong* notion of [13]. A strong threshold means that *every* set of errors of up to a certain size can be corrected, whereas a weak threshold implies that *almost all* error sets of a certain size are recoverable. However, [14] still leaves a slack of about an order of magnitude in its prediction of the error-correcting performance compared with the practical results.

The analysis of [13] and [14] are through dual certificate types of arguments for the success of the LP decoder and require codes that are based on bipartite expander graphs. A more recent work of Arora et al. uses a quite different certificate based on the primal LP problem [15]. This approach results in fairly easier computations and significantly better thresholds for LP decoding. However, the types of codes used in [15] require factor graphs with a doubly logarithmic girth rather than ones with expansion. It should be noted that similar to [14], the bounds of [15] are weak bounds; for a random set of errors of size up to a fraction of the code length (known as the weak recovery threshold), LP decoding succeeds.

A somewhat related problem to the LP decoding of linear codes is called compressed sensing. In compressed sensing an unknown real vector x of size n is to be recovered from a set of m linear measurement $y = Ax$, where $m \ll n$. This is in general infeasible, since the measurement matrix A is underdetermined and the system of equations can have infinitely many solutions. However, imposing a sparsity condition on x can make the solution unique. The unique sparse solution can be found by exhaustive search for instance, which is formulated by the following minimization program:

$$\begin{aligned} & \text{minimize } \|x\|_0 \\ & \text{subject to } Ax = y. \end{aligned} \quad (5)$$

Since (5) is NP-hard, one possible approximation is by relaxing the ℓ_0 norm of x with the closet convex norm $\|x\|_1$. Thus the ℓ_1 minimization recovery becomes:

$$\text{minimize } \|x\|_0 \quad (6)$$

$$\text{subject to } Ax = y. \quad (7)$$

(7) is a linear program and solving it requires polynomial complexity in n . For some seminal results on compressed sensing, see [17], [18], [22], [23], [25]

Recently, systematic connections between the problems of channel coding LP and compresses sensing ℓ_1 relaxation has been found [16]. In this paper, we build on those connections to improve LP decoding, and further extend the ideas of robustness and reweighted ℓ_1 minimization tin compressed sensing to channel coding LP.

IV. EXTENDED CERTIFICATE AND ROBUSTNESS OF LP DECODER

The success of LP decoder is often certified by the existence of a *dual witness* [13], [14]. Similarly, for ℓ_1 minimization in the context of compressed sensing, a dual witness certificate can guarantee that the recovery of sparse signals is successful [21]. However, for compressed sensing, people have found it easier to express the success condition in terms of the properties of the null space of the measurement matrix [22], [23], [24]. The condition is called *null space property*. The advantage of the null space interpretation, apart from better bounds, is that with proper parametrization, it can also be used to evaluate the performance of the ℓ_1 minimization in the presence of noise. This is known as the *robustness* of ℓ_1 minimization. A consequence of the robustness property is that when ℓ_1 minimization fails to recover a sparse signal, it often gives a decent approximation to it [19]. To the best of our knowledge, a similar certificate has not been introduced in the context of channel coding linear programming; one that can give the extent of the goodness of the LP optimal, in case it is not integral. In this section we introduce a property called fundamental cone property for an arbitrary code \mathcal{C} , and show that for binary symmetric channels, this is related to the robustness of the solution of the LP decoder. The robustness of LP decoders has two consequences. First, it implies tolerance to the mismatch. Second, it can be used to develop iterative schemes that improve the performance of the decoder. We will discuss these in proceeding sections. We begin by defining the fundamental cone of a code from [16].

Definition 1. Let H be a parity check matrix. Define \mathcal{J} and \mathcal{I} to be the set of rows and columns of H . Also, for each $j \in \mathcal{J}$, define $\mathcal{I}_j = \{i \in \mathcal{I} \mid H(j, i) = 0\}$. The fundamental cone \mathcal{K} , $\mathcal{K}(H)$ of H is the set of all vectors $\omega \in \mathbb{R}^n$ that satisfy

$$\omega_i \geq 0 \quad \forall 1 \leq i \leq n \quad (8)$$

$$\omega_i \leq \sum_{i' \in \mathcal{I}_j \setminus i} \omega_{i'} \quad \forall j \in \mathcal{J} \quad \forall i \in \mathcal{I}_j \quad (9)$$

$\mathcal{K}(H)$ is the minimal cone in \mathbb{R}^n that encompasses the polytope \mathcal{P} . If a vector lies on an edge of \mathcal{K} , it is called a *minimal pseudo-codeword*.

Definition 2. Let $S \subset \{1, 2, \dots, n\}$ and $C \geq 1$ be fixed. A code \mathcal{C} with parity check matrix H is said to have the fundamental cone property $FCP(S, C)$ if for every $w \in \mathcal{K}(H)$ the following holds:

$$C \|w_S\|_1 < \|w_{S^c}\|_1 \quad (10)$$

If for every index set S of size k , \mathcal{C} has the $FCP(S, C)$, then we say that \mathcal{C} has the fundamental cone property $FCP(k, C)$.

In the next lemma we show how the fundamental cone property can be used to evaluate the performance of an LP decoder, even when it fails to recover the true codeword. The key assumption is that the channel is a BSC. Due to the similarity of the proof with the corresponding compressed sensing statement, we omit the proof here, but it can be found in [28].

Lemma 1. Let \mathcal{C} be a code that has the $FCP(S, C)$ for some index set S and some $C \geq 1$. Suppose that a codeword $x^{(c)}$ from \mathcal{C} is transmitted through a BSC channel, and the received codeword is $x^{(r)}$. If the pseudocodeword $x^{(p)}$ is the output of LP decoder for the received codeword $x^{(r)}$, then the following holds:

$$\|x^{(p)} - x^{(c)}\|_1 < 2 \frac{C+1}{C-1} \|(x^{(r)} - x^{(c)})_{S^c}\|_1 \quad (11)$$

An asymptotic case of Lemma 1 for $C \rightarrow 1$ is in fact equivalent to the LP success condition. Namely, let S be the index set of the flipped bits in the transmitted codeword, i.e. the set of bits that differ in $x^{(r)}$ and $x^{(c)}$. If for some $C > 1$ the $FCP(S, C)$ holds for the code \mathcal{C} , then Lemma 1 implies that LP decoding can successfully recover the original codeword. Now let us say that the set of errors (flipped bits) is slightly larger than S , and does include S . Then the vector $(x^{(r)} - x^{(c)})_{S^c}$ has a few (but not too many) nonzero entries. Therefore, even if the LP decoder output $x^{(p)}$ is not equal to the actual codeword, it is still possible to give an upper bound on its ℓ_1 distance to the unknown codeword. This is the property that we recognize as the robustness of LP decoder, and is quantified by the $FCP(S, C)$, when $C > 1$. We consider two notions of robustness Strong robustness means that for every set S of up to some cardinality, the FCP condition holds. Weak robustness on the other hand deals with almost all sets S of up to a certain size. In the next section we do a thorough analysis of LP robustness for two categories of codes; expander codes and codes with $\Omega(\log \log n)$ girth. For these two categories, rigorous analysis has been done on the performance of LP decoders in [13], [14] and [15] respectively. Afterwards, we discuss how the robustness results can help improve the LP decoder or make it tolerant to mismatch.

V. ANALYSIS OF LP ROBUSTNESS

If there is a certificate for the success of LP decoder, it can be often extended to guarantee that the LP decoder is robust, namely that the FCP condition is satisfied for some $C > 1$. Only by carefully re-examining the analysis of LP decoder, one might be able to do such a generalization. This is the main target of this section. We consider three major analysis of LP decoders in the literature. The first one is by Feldman et. al [13], which uses a dual witness type of argument to certify the success of LP, and is based on expander graphs. The second one is by Daskalakis et al. [14] which again

considers linear programming decoding in expander codes. Specifically, [14] analyzes the dual of LP and finds a simple combinatorial condition for the dual value to be zero (implying that the LP decoder is successful). The condition is basically the existence of a so-called *hyperflow* from the set of flipped bits to unflipped bits. The existence of a valid hyperflow can be secured by the presence of the so-called (p, q) -matchings, and the whole premise of using expander codes is to verify that with high probability (p, q) -matchings exist. The main difference between this analysis and that of Feldman et al. is the probabilistic nature of the arguments in [14], which account for weak recovery thresholds.

A third analysis of the LP decoder was done by Arora et al., [15], which is based on factor graphs with a girth logarithmic in the number of variable nodes. Unlike previous dual feasibility arguments, the authors in [15] introduce a certificate in the primal domain, which is of the following form. If in the primal LP problem, the value of the objective function for the original codeword is smaller than its value for all vectors within a local deviation from the original codeword, then LP decoder succeeds. Local deviations are defined by weighted minimal local trees whose induced subgraph does not have a cycle.

A. Strong LP Robustness for Expander Codes

Strong thresholds of LP decoding for expander codes are derived in [13]. To show that the transmitted codeword is the solution to (3), namely the LP optimal, when a subset of the bits are flipped, a set of feasible dual variables are found that satisfy the following conditions. Suppose the factor graph of \mathcal{C} is denoted by $\mathcal{G} = (X_v, X_c, \mathcal{E})$. We may also assume without loss of generality that the all zero codeword was transmitted. A set of feasible dual variables is defined as follows ([13])

Definition 3. For an error set S , a set of feasible dual variables is a labeling of the edges of the factor graph \mathcal{G} , say $\{\tau_{ij} \mid v_i \in X_v, c_j \in X_c\}$, where the following two conditions are satisfied:

- i) For every check node $c_j \in X_c$ and every two disjoint neighbors of c_j like $v_i, v_{i'} \in N(j)$, we have $\tau_{ij} + \tau_{i'j} \geq 0$.
- ii) For every variable node $v_i \in X_v$, we have $\sum_{c_j \in N(v_i)} \tau_{ij} \leq \gamma_i$.

We show that a generalized set of dual feasible variables can be used to derive LP robustness. To this end, we show that the existence of a set of feasible dual variables implies the FCP condition. The proof of the following lemma is omitted due to lack of space, but can be found in the detailed version of this paper [28].

Lemma 2. Suppose that a set of dual variables satisfies the feasibility conditions (Definition 3) for an arbitrary log-likelihood vector γ . Then for every vector $\mathbf{w} \in \mathcal{K}(\mathcal{C})$, the following holds

$$\sum_{1 \leq i \leq n} \gamma_i w_i > 0. \quad (12)$$

A special case of Lemma 2 is when the channel is a BSC, and a set S of the bits have been flipped. We can also assume without

loss of generality that the all zero codeword was transmitted. Then Lemmas 1 and 2 imply that if a dual feasible set exists, then LP decoder succeeds, which is the conclusion of [13]. In this case the log-likelihood vector γ takes the value -1 over the set S and value 1 over the set S^c . Let us now define a new likelihood vector γ' by

$$\gamma' = \begin{cases} -C & i \in S \\ 1 & i \in S^c \end{cases}, \quad (13)$$

for some $C > 1$. If a dual feasible set exists that satisfies the feasibility condition for γ' , then it follows that $\text{FCP}(S, C)$ holds. Knowing this and pursuing an argument very similar to [13] for the construction of dual feasible in expander codes, we are able to prove the following lemma, the proof of which is omitted for brevity.

Theorem 1. Let \mathcal{G} be the factor graph of a code \mathcal{C} of length n and rate $R = \frac{m}{n}$, and let $\delta > 2/3 + 1/d_v$. If \mathcal{G} is a bipartite $(\alpha n, \delta d_v)$ expander, then \mathcal{C} has $\text{FCP}(t, C)$, where $t = \frac{3\delta-2}{2\delta-1}\alpha$ and $C = \frac{2\delta-1}{2\delta-1-1/d_v}$. This means that for every set S of size t , $\text{FCP}(t, C)$ holds.

Basically, [13] shows that if the conditions of Theorem 1 are satisfied, then LP succeeds, namely that $\text{FCP}(t, 1)$ holds. However Theorem 1 asserts that, in addition, a strong robustness holds, namely $\text{FCP}(t, C)$ for some $C > 1$.

B. Weak LP Robustness for Expander Codes

We show that for random expander codes a probabilistic analysis similar to the dual witness analysis of [14] can be used to find the extents of the fundamental cone property for expander codes, in a weak sense. We rely on the matching arguments of [14], with appropriate justifications. The following definition is given in [14].

Definition 4. For nonnegative integers p and q , and a set F of variable nodes, a (p, q) -matching on F is defined by the following conditions:

- (a) each bit $v_i \in F$ must be matched with p distinct check nodes, and
- (b) each variable node $v_{i'} \in F^c$ must be connected with

$$X_{i'} := \max\{q - d_v + Z_{i'}, 0\} \quad (14)$$

checks nodes from the set $N(F)$, that are different from the check nodes that the nodes in F are matched to, where $Z_{i'}$ is defined as $Z_{i'} := |N(i') \cap N(F)|$.

We prove the following lemma that relates the existence of a (p, q) -matching to the fundamental cone property of a code \mathcal{C} . We omit the proof for brevity, but it can be found in [28].

Lemma 3. Let \mathcal{C} be a code of rate R with a bipartite factor graph \mathcal{G} , where every variable node has degree d_v . Let S be a subset of the variable nodes of \mathcal{G} . If a (p, q) -matching on S exists, then \mathcal{C} has the $\text{FCP}(S, \frac{2p-d_v}{d_v-q})$.

[14] provides a probabilistic tool for the existence of (p, q) -matchings in regular bipartite expander graphs, which helps answer the question of how large an error set LP decoding can fix. For example, for a random LDPC(8,16) code, the

probabilistic analysis implies that a fraction 0.002 of errors is recoverable using LP decoder. However, taking the specifications of the matching that leads to this conclusion and applying Lemma 3, it turns out that for an error set of size $0.002n$, the robustness factor is at least $C = 1.3$, i.e the code has $\text{FCP}(0.002n, 1.3)$.

C. Weak LP Robustness for Codes with $\Omega(\log \log(n))$ Girth

Recall that $\mathcal{G} = (X_v, X_c, \mathcal{E})$ is used to denote the factor graph of the parity check matrix H (or of code C), where X_v and X_c are the sets of variable and check nodes respectively and \mathcal{E} is the set of edges. Also recall that the girth of \mathcal{G} is defined as the size of the shortest cycle in \mathcal{G} . Without loss of generality, we assume that $X_v = \{v_1, v_2, \dots, v_n\}$, where v_i is the variable node corresponding to the i th bit of the codeword. Let $T \leq \frac{1}{4}\text{girth}(\mathcal{G})$ be fixed. The following notions are defined in [15].

Definition 5. A tree \mathcal{T} of height $2T$ is called a *skinny subtree* of \mathcal{G} , if it is rooted at some variable node v_{i_0} , for every variable node v in \mathcal{T} all the neighboring check nodes of v in \mathcal{G} are also present in \mathcal{T} , and for every check node c in \mathcal{T} exactly two neighboring variable nodes of c in \mathcal{G} are present in \mathcal{T} .

Definition 6. Let $w \in [0, 1]^T$ be a fixed vector. A vector $\beta^{(w)}$ is called a *minimal T -local deviation*, if there is a skinny subtree of \mathcal{G} of height $2T$, say \mathcal{T} , so that for every variable node v_i $1 \leq i \leq n$,

$$\beta_i^{(w)} = \begin{cases} w_{h(i)} & \text{if } v_i \in \mathcal{T} \setminus \{v_{i_0}\} \\ 0 & \text{otherwise} \end{cases}.$$

Where $h_i = \frac{1}{2}d(v_{i_0}, v_i)$.

The key to the derivations of [15] is the following lemma:

Lemma 4 (Lemma 1 of [15]). *For any vector $z \in \mathcal{P}$, and any positive vector $w \in [0, 1]^T$, there exists a distribution on the minimal T -local deviations $\beta^{(w)}$, such that*

$$\mathbb{E}\beta^{(w)} = \alpha z,$$

where $0 < \alpha \leq 1$.

Lemma 4 has the following interpretation. If a linear property holds for all minimal T -local deviations (e.g. $f(\beta^{(w)}) \geq 0$, where $f(\cdot)$ is linear), then it also holds for all pseudocodewords ($f(z) \geq 0 \forall z \in \mathcal{P}$). Interestingly enough, the robustness of LP decoding for a given set of bit flips S has a linear certificate, namely $\text{FCP}(S, C)$ ¹. In other words, if we define:

$$f_C^{(S)}(x) = \sum_{i \in S^c} x_i - C \sum_{i \in S} x_i,$$

then $\text{FCP}(S, C)$ holds, if and only if $f_1^{(S)}(z) \geq 0$ for every pseudocodeword $z \in \mathcal{P}$. Therefore, according to Lemma 4, it suffices that the condition be true for all T -local deviations. Furthermore, for arbitrary $C > 1$, if $f_C^{(S)}(\beta^{(w)}) \geq 0$ for all

¹Note that this is only true for binary symmetric channels

minimal T -local deviations $\beta^{(w)}$, then it follows that the code has the $\text{FCP}(S, C)$ property. This simple observation helps us extend the probabilistic analysis of [15] to robustness results for LP decoding. The resulting key theorem is mentioned below, the proof is omitted for brevity, but can found in the online version of this paper [28]. In order to state the theorem, first we define η_C to be a random variable that takes the value $-C$ with probability p and value 1 with probability $1 - p$. Also, define the sequences of random variables X_i, Y_i , $i \geq 0$, in the following way:

$$\begin{aligned} Y_0 &= \eta_C, \\ X_i &= \min\{Y_i^{(1)}, \dots, Y_i^{(d_c-1)}\} \quad \forall i > 0, \\ Y_i &= 2^i \eta_C + X_{i-1}^{(1)} + \dots + X_{i-1}^{(d_v-1)} \quad \forall i > 0, \end{aligned} \quad (15)$$

Where $X^{(j)}$ s are independent copies of a random variable X .

Theorem 2. *Let $0 \leq p \leq 1/2$ be the probability of bit flip, and S be the random set of flipped bits. If for some $j \in \mathbb{N}$ the following holds*

$$c = \left(\min_{t \geq 0} \mathbb{E} e^{-tX_j} \right) (d_c - 1) \frac{C+1}{C} \left(\frac{Cp}{1-p} \right)^{1/(C+1)} (1-p) < 1$$

Then with probability at least $1 - nc'c^{d_v(d_v-1)^{T-1}}$ the code C has the $\text{FCP}(S, C)$, where T is any integer with $j \leq T < 1/4\text{girth}(\mathcal{G})$, and c' is a positive constant.

For $d_c = 6$ and $d_v = 3$, a lower bound for the curve that results from Theorems 2 for C is plotted in Figure 1.

VI. IMPLICATIONS OF LP ROBUSTNESS

A. Mismatch Tolerance

One of the direct consequences of the robustness of LP decoding is that if there is a slight mismatch in the formulation of the LP decoder, its performance does not degrade significantly. More formally, suppose that due to noise, quantization or some other factor, a mismatched log-likelihood vector $\gamma' = \gamma + \Delta\gamma$ is used in the LP implementation. We refer to such a decoder as a *mismatched LP decoder*. Since the channel is BSC, the entries of γ all have the same amplitude g . We also define $\delta = \max_i |\Delta\gamma_i|$, and assume that $\delta < g$. We can prove the following theorem.

Theorem 3. *Suppose that S is the set of bit errors. Let $C = \frac{g+\delta}{g-\delta}$. If C has $\text{FCP}(S, C)$, then the mismatched LP decoder can fix the errors.*

Proof: We assume without loss of generality that the all zero codeword is transmitted. We show that if $\text{FCP}(S, C)$ holds, then the all zero codeword is the minimum cost vector in the polytope \mathcal{P} . More generally, suppose w is a nonzero vector in the fundamental code \mathcal{K} . We begin with the definition of $\text{FCP}(S, C)$ and write

$$-C \sum_{i \in S} w_i + \sum_{i \in S^c} w_i > 0. \quad (16)$$

Multiply both sides by $(g - \delta)$:

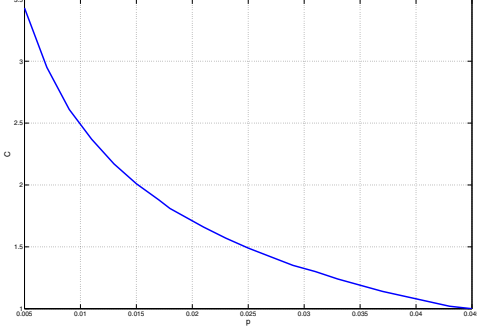


Fig. 1: Approximate upper bound for the robustness factor C as a function of error probability p for $d_c = 6$ and $d_v = 3$, based on Theorem 2.

$$-\sum_{i \in S} (g + \delta)w_i + \sum_{i \in S^c} (g - \delta)w_i > 0 \quad (17)$$

We also know from the definition of δ that $\gamma'_i > (g - \delta)$ for $i \in S^c$, and $\gamma'_i > -g - \delta$ for $i \in S^c$, and that $w \geq 0$. Therefore

$$-\sum_{i \in S \cup S^c} \gamma'_i w_i > 0 \quad (18)$$

which proves that the all zero codeword is the unique minimum cost solution of the mismatched LP.

B. Pseudocodewords and High Error Rate Subsets

We showed in Section IV that for an appropriate code \mathcal{C} , even when LP decoder fails to recover an actual codeword from the output of a BSC, the ℓ_1 distance between the obtained pseudocodeword and the actual codeword can be decently bounded. We now show that this property allows us to find a *high error* rate subset of the bits, namely a subset of the bits over which the fraction of errors is significantly larger than the fraction of errors in the entire received codeword. One can then put additional soft or hard constraints on that *importance* subset, and run a constrained linear program or other post processing algorithms. This forms the idea for the iterative LP decoding algorithm that we will propose in Section VII.

Consider a code \mathcal{C} of length n and rate R . Consider a codeword $x^{(c)}$ from \mathcal{C} transmitted through a BSC, and suppose that a set K of the bits get flipped, where the cardinality of K is $(1 + p^*)\epsilon n$ for some $0 < p^* < 1$ and $\epsilon > 0$. Denote the received vector by $x^{(r)}$. We want to consider the case where LP fails, so the LP minimal $x^{(p)}$ is a fractional pseudocodeword. However, the size of the error set is not too larger than the correctable size. In other words, we assume that for some subset $K_1 \subset K$ of size p^*n , the code has $\text{FCP}(K_1, C)$, and for some $C > 1$. We show in the next lemma that if we look at the index set of the largest k entries of the vector $x^{(r)} - x^{(p)}$ in magnitude, the overlap between this set and K can be quite significant. The following theorem formalized this claim.

Theorem 4. Suppose that a codeword $x^{(c)}$ is transmitted through a BSC, and the output $x^{(r)}$ differs from the input in a set K of the bits with $|K| = p^*(1 + \epsilon)n$, for some $0 < p^* < 1$ and $\epsilon > 0$. Also, suppose that for a subset $K_1 \subset K$ of size p^*n , the $\text{FCP}(K_1, C)$ holds, for some $C > 1$, and that the LP minimal is the pseudocodeword $x^{(p)}$. If L is the set of the $p^*(1 + \epsilon)n$ largest entries of the vector $x^{(r)} - x^{(p)}$ in magnitude, then the fraction of errors in $x^{(r)}$ over the set L is at least $1 - 2\frac{C+1}{C-1}\epsilon$.

Before proving this theorem, we bring the following definition and lemma.

Definition 7. Let $x \in \mathbb{R}^n$ be a k -sparse vector. For $\lambda > 0$, We define $W(x, \lambda)$ to be the size of the largest subset of nonzero entries of x that has a ℓ_1 norm less than or equal to λ .

$$W(x, \lambda) := \max\{|S| \mid S \subseteq \text{supp}(x), \|x_S\|_1 \leq \lambda\} \quad (19)$$

The following Lemma is proven in [19].

Lemma 5 (Lemma 1 of [19]). Let x be a k -sparse vector and \hat{x} be another vector. Also, let K be the support set of x and L be the k -support set of \hat{x} , namely the set of k largest entries of \hat{x} . If $d = \|x - \hat{x}\|_1$, then

$$|K \cap L| \geq k - W(x, d) \quad (20)$$

Proof of Theorem 4:

Define $k = p^*(1 + \epsilon)n$, and apply Lemma 5 to the k -sparse vector $x^{(r)} - x^{(c)}$, and the vector $x^{(p)} - x^{(r)}$. If L is the index set of the largest k entries of $x^{(p)} - x^{(r)}$ in magnitude, then from Lemma 5 we have

$$|K \cap L| \geq k - W(x^{(r)} - x^{(c)}, \Delta) \quad (21)$$

Where $\Delta = \|x^{(c)} - x^{(p)}\|_1$. Since $\|x^{(r)} - x^{(c)}\|_1$ has only ± 1 nonzero entries, (21) can be written as

$$|K \cap L| \geq k - \|x^{(c)} - x^{(p)}\|_1 \quad (22)$$

We use the inequality in (11) to further lower bound the right hand side of (22). Recall that $K_1 \subset K$ is such that \mathcal{C} has $\text{FCP}(K_1, C)$. Therefore, we can write:

$$|K \cap L| \geq k - 2\frac{C+1}{C-1}\|(x^{(r)} - x^{(c)})_{K_1^c}\|_1 \quad (23)$$

$$= k - 2\frac{C+1}{C-1}(k - p^*n) \quad (24)$$

Dividing both sides by $|K| = k$, we conclude that at least a fraction $1 - 2\frac{C+1}{C-1}\epsilon$ of the set L are flipped bits. ■

VII. ITERATIVE REWEIGHTED LP ALGORITHM AND IMPROVED STRONG THRESHOLD

We briefly define different recovery thresholds for LP decoding first. In previous sections, we carelessly referred to the weak and strong thresholds, and the corresponding weak and strong robustness. In general, the actual weak and strong thresholds for a given classes of linear codes might be unknown, and all the analysis of the thresholds in the

literature only provide lower bounds on these quantities. For expander codes for instance, the size of the error set that can be recovered via LP can be lower bounded by the size of the set for which a dual witness exists [13], [14]. Since a dual witness is only a sufficient condition for the success of LP decoding, the actual thresholds might be higher. However, to date, the best achievable thresholds for LP decoding for expander codes are those given by the dual feasibility. Therefore, we also consider thresholds associated with those limits, namely the provable thresholds. Specifically, we define the following four thresholds for LP decoding on a given code C that has a regular variable and check degrees d_v and d_c .

Definition 8 (Recovery thresholds). *Strong recovery threshold is denoted by p_s^* , and is defined as the largest fraction such that every set of size p_s^*n is recoverable via LP decoding. Weak recovery threshold is denoted by p_w^* , and it means that almost all sets of size p_w^*n is recoverable via LP. We define p_{sd}^* to be the maximum provable strong threshold achieved by a dual feasible, [13]. Similarly, p_{wd}^* is the provable weak threshold, i.e. for almost all sets of size p_{wd}^*n , a dual feasible ([14]) exist.*

By looking at the deviation of the LP optimal (pseudocodeword) and the received vector, we can identify a subset of bits that has a high probability of bit flip, which we call a High Error Rate (HER) subset. If the number of flipped bits in one set is significantly higher than the other set, then we can incorporate this imbalancedness by using a weighted LP scheme. This is the main idea for the following iterative algorithm.

We now introduce the reweighted LP decoding algorithm.

Algorithm 1.

- 1) Run LP decoding. If the output is integral terminate, otherwise proceed.
- 2) Take the fractional pseudocodeword $x^{(p)}$ from the LP decoder, and construct the deviation vector $x^{(d)} = x^{(r)} - x^{(p)}$.
- 3) Sort the entries of $x^{(d)}$ in terms of absolute value, and denote by L the index set of the *smallest* pn entries.
- 4) solve the following weighted LP:

$$\min_{x \in \mathcal{P}} \lambda_1 \|(x - y)_L\|_1 + \lambda_2 \|(x - y)_{L^c}\|_1 \quad (25)$$

where λ_1 and λ_2 , where $\lambda_1 < 0$ and $\lambda_2 > 0$ are fixed parameters.

Algorithm 1 is only twice as complex as LP decoding, and is still polynomial time. We next prove that algorithm 1 has a strictly improved provable strong recovery threshold than the strong dual feasibility threshold p_{sd}^* . Recall the definition of p_{sd}^* from Definition 8. p_{sd}^* is basically the provable strong threshold of LP decoding based on the analysis of [13]. The proof of the following theorem has been omitted due to lack of space, but can be found in the detailed version of this paper, [28].

Theorem 5. *For any code C , there exists an $\epsilon_1 > 0$ and $\epsilon_2 > 0$, $\lambda_1 < 0$ and $\lambda_2 > 0$ so that for every error set of size $(1 + \epsilon_1)p_{sd}^*$, and almost all error sets of size $(1 + \epsilon_2)p_{wd}^*$*

*can be corrected by Algorithm 1. Furthermore, if $(C - 1)/\epsilon$ is large enough, where C is the robustness factor for sets of size $(1 - \epsilon)p_s^*n$ (or $(1 - \epsilon)p_w^*n$), then the strong (weak) threshold on Algorithm 1 is at least $(1 + \epsilon)p_s^*$ (or $(1 + \epsilon)p_w^*$).*

VIII. SIMULATIONS

We have implemented Algorithm 1 on a random LPDC code of size $n = 1000$ and rate $R = 3/4$ and have compared the results with other existing methods. The variable node degree is $d_v = 3$, and thus, $d_c = 4$. The algorithm is compared with the mixed integer method of Draper and Yedidia [26], and the random facet guessing algorithm of [27]. The mixed integer algorithm basically reruns LP by setting integer constraints on a small subset of “least certain” bits, namely the positions where the LP minimal pseudocodeword entries are closest to 0.5. We have taken the size of the constrained subset to be $M = 5$, which means the number of extra iterations is 32. We also choose to run 20 more extra random iterations for facet guessing. In random facet guessing, a face (facet) of the polytope \mathcal{P} is selected at random, among all the faces on which the LP minimal pseudocodeword does not reside. Then, LP decoder is re-run with the additional constraint that the solution is on the selected face. In contrast Algorithm 1 has only one extra iteration. The algorithms are run in MATLAB where LP decoder is implemented via the cvx toolbox [29]. We have plotted the BER curves versus the probability of error p in Figure 2. For Algorithm 1, for each p , we have experimentally found the optimal λ_1 and λ_2 by choosing the values that on average result in the best performance. For most of the cases the chosen values were in the ranges $-3 \leq \lambda_1 \leq -0.5$ and $1 \leq \lambda_2 \leq 3$. When n becomes larger, the BER curves for all iterative methods collapse into the same curve as the LP curve, except for the reweighted LP algorithm.

REFERENCES

- [1] J. Feldman, “Decoding Error-Correcting Codes via Linear Programming”, PhD thesis, MIT, 2003.
- [2] J. Feldman, M. J. Wainwright, and D. R. Karger, “Using linear programming to decode binary linear codes,” IEEE Transactions on Information Theory, 51(3):954-972, 2005.
- [3] J. Feldman, D. R. Karger and M. J. Wainwright, “Linear Programming-Based Decoding of Turbo-Like Codes and its Relation to Iterative Approaches,” Proc. 40th Annual Allerton Conf. on Communication, Control, and Computing Oct. 2002.
- [4] M. J. Wainwright, T. S. Jaakkola and A. S. Willsky, “MAP estimation via agreement on (hyper)trees: Message-passing and linear programming approaches,” Proc. Allerton Conference on Communication, Control and Computing Oct. 2002.
- [5] R. Koetter and P. O. Vontobel, “Graph-covers and iterative decoding of finite length codes,” Proc. 3rd International Symp. on Turbo Codes, Sep. 2003.
- [6] P. O. Vontobel and R. Koetter, “Towards low-complexity linear-programming decoding,” Proc. Int. Conf. on Turbo Codes and Related Topics, Munich, Germany, Apr. 2006.
- [7] M. H. Taghavi and P. H. Siegel, “Adaptive linear programming decoding,” IEEE Int. Symposium on Information Theory, Seattle, WA, July 2006.
- [8] M. P. C. Fossorier, “Iterative reliability-based decoding of low-density parity check codes,” IEEE Transactions on Information Theory, May 2001, pp:908-917.

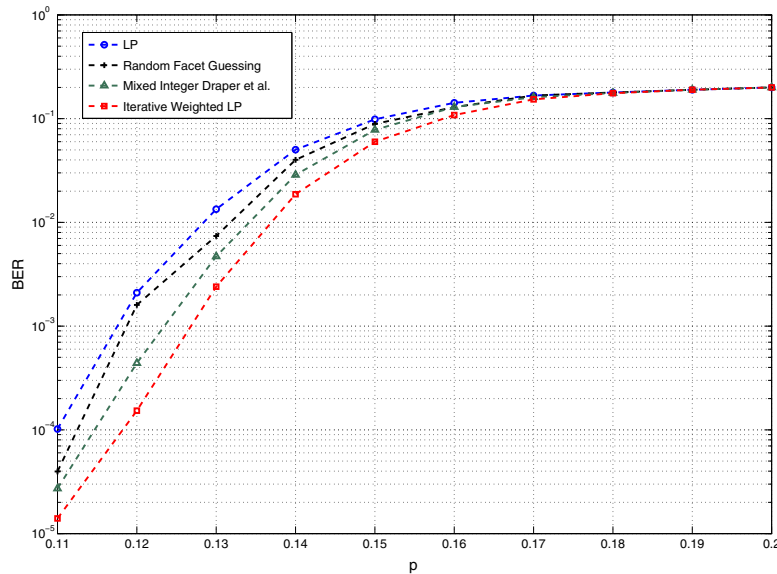


Fig. 2: BER curves as a function of channel flip probability p , for LP decoding and different iterative schemes; random facet guessing of [27], mixed integer method of [26], and the suggested iterative reweighted LP of Algorithm 1. The code is a random LDPC(3,4) of length $n = 100$.

- [9] H. Pishro-Nik and F. Fekri, "On Decoding of LDPC Codes over the Erasure Channel," *IEEE Trans. Inform. Theory*, Vol. 50, pp:439-454 2004.
- [10] K. Yang, J. Feldman and X. Wang "Nonlinear programming approaches to decoding low-density parity-check codes," *IEEE J. Sel. Areas in Communication*, Vol. 24 NO. 8, pp: 1603-1613, Aug. 2006.
- [11] M. Chertkov and V. Y. Chernyak, "Loop calculus helps to improve belief propagation and linear programming decoding of LDPC codes", Allerton Conference on Communications, Control and Computing, Monticello, IL, Sep. 2006.
- [12] M. J. Wainwright and T. S. Jaakkola and A. S. Willsky, "Exact MAP estimates via agreement on (hyper)trees: Linear programming and message-passing," *IEEE Trans. Information Theory*, Vol. 51, NO. 11 pp: 3697-3717, Nov. 2005.
- [13] J. Feldman, T. Malkin, R. A. Servedio, C. Stein, and M. J. Wainwright, "LP decoding corrects a constant fraction of errors", In *Proc. IEEE International Symposium on Information Theory*, 2004.
- [14] C. Daskalakis, A. G. Dimakis, R. M. Karp and M. J. Wainwright, "Probabilistic Analysis of Linear Programming decoding", *IEEE Transactions on Information Theory*, Volume 54, Issue 8, Aug. 2008.
- [15] S. Arora, D. Steurer, and C. Daskalakis, "Message-Passing Algorithms and Improved LP Decoding", *ACM STOC* 2009.
- [16] A.G. Dimakis and P. Vontobel, "LP Decoding meets LP Decoding: A Connection between Channel Coding and Compressed Sensing", Allerton 2009.
- [17] D. Donoho, "High-dimensional centrally symmetric polytopes with neighborliness proportional to dimension," *Discrete and Computational Geometry*, 102(27), pp. 617-652 2006, Springer.
- [18] D. Donoho and J. Tanner, "Neighborliness of randomly-projected simplices in high dimensions," *Proc. National Academy of Sciences*, 102(27), pp. 9452-9457, 2005.
- [19] A. Khajehnejad, W. Xu, S. Avestimehr, B. Hassibi, "Improved Sparse Recovery Thresholds with Two-Step Reweighted ℓ_1 Minimization", *ISIT* 2010.
- [20] E. J. Candès, M. B. Wakin, and S. Boyd, "Enhancing Sparsity by Reweighted ℓ_1 Minimization", *Journal of Fourier Analysis and Applications*, 14(5), pp. 877-905, special issue on sparsity, December 2008.
- [21] E. J. Candès and T. Tao, "Decoding by linear programming", *IEEE Trans. Inform. Theory*, 51 4203-4215
- [22] D.L. Donoho and X. Huo, "Uncertainty principles and ideal atomic decomposition", *IEEE Transactions on Information Theory*, 47(7):2845-2862, 2001.
- [23] M. Stojnic, W. Xu, and B. Hassibi, "Compressed sensing - probabilistic analysis of a null-space characterization" *IEEE International Conference on Acoustic, Speech and Signal Processing, ICASSP 2008*
- [24] A. Cohen, W. Dahmen, and R. DeVore, "Compressed sensing and best k-term approximation", *Journal of the American Mathematical Society*, Volume 22, Number 1, January 2009, Pages 211-231
- [25] W. Xu and B. Hassibi, "On Sharp Performance Bounds for Robust Sparse Signal Recoveries", accepted to *the International Symposium on Information Theory 2009*.
- [26] S.C. Draper, J.S. Yedidia, Y. Wang, "ML Decoding via Mixed-Integer Adaptive Linear Programming", *IEEE International Symposium on Information Theory (ISIT)*, June 2007 (ISIT 2007, TR2007-022).
- [27] A. G. Dimakis, A. A. Gohari and M. Wainwright, "Guessing Facets: Polytope Structure and Improved LP Decoder", *IEEE Transactions on Information Theory*, Volume 55, Issue 8, Aug. 2009,
- [28] A. Khajehnejad, A. G. Dimakis, B. Hassibi, W. Bradley, "Robustness of LP and Iterative Reweighted LP Decoding", preprint 2010.
- [29] cvx toolbox webpage <http://cvxr.com/cvx/>.